N 11

```
VVV         VVV  MMM         MMM     SSSSSSSSSSSS  LLL                 IIIIIIIII     BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSSSSS  LLL                 IIIIIIIII     BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSSSSS  LLL                 IIIIIIIII     BBBBBBBBBBBB
VVV         VVV  MMMMMM   MMMMMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMMMMM   MMMMMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMMMMM   MMMMMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM   MMM   MMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM   MMM   MMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM   MMM   MMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM         MMM     SSSSSSSSS     LLL                    III        BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSS     LLL                    III        BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSS     LLL                    III        BBBBBBBBBBBB
VVV         VVV  MMM         MMM              SSS  LLL                    III        BBB         BBB
VVV         VVV  MMM         MMM              SSS  LLL                    III        BBB         BBB
VVV         VVV  MMM         MMM              SSS  LLL                    III        BBB         BBB
   VVV   VVV     MMM         MMM              SSS  LLL                    III        BBB         BBB
   VVV   VVV     MMM         MMM              SSS  LLL                    III        BBB         BBB
   VVV   VVV     MMM         MMM              SSS  LLL                    III        BBB         BBB
      VVV        MMM         MMM  SSSSSSSSSSSSS    LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB
      VVV        MMM         MMM  SSSSSSSSSSSSS    LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB
      VVV        MMM         MMM  SSSSSSSSSSSSS    LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB
```

```
**FILE**ID**LIBASSIGN                              G 7

LL          IIIIII  BBBBBBBB      AAAAAA      SSSSSSSS    SSSSSSSS  IIIIII    GGGGGGGG  NN      NN
LL          IIIIII  BBBBBBBB      AAAAAA      SSSSSSSS    SSSSSSSS  IIIIII    GGGGGGGG  NN      NN
LL            II    BB      BB  AA      AA  SS          SS            II    GG          NN      NN
LL            II    BB      BB  AA      AA  SS          SS            II    GG          NN      NN
LL            II    BB      BB  AA      AA  SS          SS            II    GG          NNNN    NN
LL            II    BB      BB  AA      AA  SS          SS            II    GG          NNNN    NN
LL            II    BBBBBBBB    AA      AA    SSSSSS        SSSSSS        II    GG          NN  NN  NN
LL            II    BBBBBBBB    AA      AA    SSSSSS        SSSSSS        II    GG          NN  NN  NN
LL            II    BB      BB  AAAAAAAAAA          SS          SS    II    GG  GGGGGG  NN    NNNN
LL            II    BB      BB  AAAAAAAAAA          SS          SS    II    GG  GGGGGG  NN    NNNN
LL            II    BB      BB  AA      AA          SS          SS    II    GG      GG  NN      NN  ....
LL            II    BB      BB  AA      AA          SS          SS    II    GG      GG  NN      NN  ....
LLLLLLLLLL  IIIIII  BBBBBBBB    AA      AA  SSSSSSSS    SSSSSSSS    IIIIII    GGGGGG    NN      NN  ....
LLLLLLLLLL  IIIIII  BBBBBBBB    AA      AA  SSSSSSSS    SSSSSSSS    IIIIII    GGGGGG    NN      NN  ....

LL          IIIIII    SSSSSSSS
LL          IIIIII    SSSSSSSS
LL            II    SS
LL            II    SS
LL            II    SS
LL            II    SS
LL            II      SSSSSS
LL            II      SSSSSS
LL            II            SS
LL            II            SS
LL            II            SS
LL            II            SS
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS
```

Table of contents

```
    (2)        44        HISTORY         ; Detailed Current Edit History
    (3)        50        DECLARATIONS
    (4)        76        LIB$ASSIGN -
```

```
0000         .TITLE  LIB$ASSIGN                      ; Assign channel to device
0000         .IDENT  'V04-000'
0000    ;
0000    ;****************************************************************************
0000    ;*                                                                          *
0000    ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000    ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000    ;*  ALL RIGHTS RESERVED.                                                    *
0000    ;*                                                                          *
0000    ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000    ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0000    ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000    ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0000    ;*  TRANSFERRED.                                                            *
0000    ;*                                                                          *
0000    ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0000    ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    ;*  CORPORATION.                                                            *
0000    ;*                                                                          *
0000    ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0000    ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000    ;*                                                                          *
0000    ;*                                                                          *
0000    ;****************************************************************************
0000    ;
0000    ;
0000    ; FACILITY: Language-independent support library
0000    ;++
0000    ; ABSTRACT:
0000    ;
0000    ;
0000    ;--
0000    ;
0000    ; VERSION: 1
0000    ;
0000    ; HISTORY:
0000    ;
0000    ; AUTHOR:
0000    ;
0000    ;       Herb Jacobs             August 1981
0000    ;
```

```
0000     44          .SBTTL  HISTORY            ; Detailed Current Edit History
0000     45
0000     46 ; EDIT HISTORY:
0000     47 ;
0000     48 ; 1-001 -
```

```
    0000          .SBTTL  DECLARATIONS
    0000
    0000  ;
    0000  ; INCLUDE FILES:
    0000  ;
    0000          $FABDEF
    0000          $NAMDEF
    0000  ;
    0000  ; EXTERNAL SYMBOLS:
    0000  ;
    0000
    0000  ;
    0000  ; MACROS:
    0000  ;
    0000
    0000  ;
    0000  ; PSECT DECLARATIONS:
    0000  ;
    0000
00000000          .PSECT  _LIB$CODE       PIC, SHR, LONG, EXE, NOWRT
    0000
    0000  ;
    0000  ; EQUATED SYMBOLS:
    0000  ;
    0000
```

```
                              0000          .SBTTL  LIB$ASSIGN -
                              0000
                              0000  ;++
                              0000  ; FUNCTIONAL DESCRIPTION:
                              0000  ;
                              0000  ;       These routines provide a general mechanism to get a channel assigned
                              0000  ;       to the associated device with a file specification.  The channel
                              0000  ;       assigned is always to the device, never to the file.  The purpose
                              0000  ;       of these routines are to successfully accomplish up cased recursive
                              0000  ;       logical name translation and succeed with process permanent files.
                              0000  ;       Since these routines use RMS as a basis, compatibility is assured
                              0000  ;       with all working programs.
                              0000  ;
                              0000  ; CALLING SEQUENCE:
                              0000  ;
                              0000  ;       ret = LIB$ASSIGN        ( filedesc, chan)
                              0000  ;       ret = LIB$ASSIGN_FILE   ( filedesc, chan)
                              0000  ;       ret = LIB$ASSIGN_DEV    ( filedesc, chan)
                              0000  ;
                              0000  ; INPUT PARAMETERS:
                              0000  ;
                              0000  ;       FILEDESC - Address of character string descriptor describing input
                              0000  ;
                              0000  ; IMPLICIT INPUTS:
                              0000  ;
                              0000  ;       NONE
                              0000  ;
                              0000  ; OUTPUT PARAMETERS:
                              0000  ;
                              0000  ;       CHAN -  Address of word to recieve channel number assigned
                              0000  ;
                              0000  ; IMPLICIT OUTPUTS:
                              0000  ;
                              0000  ;       NONE
                              0000  ;
                              0000  ; COMPLETION CODES:
                              0000  ;
                              0000  ;       SS$_NORMAL      - success
                              0000  ;       All other completion codes are generated as RMS or SS codes.
                              0000  ;
                              0000  ; SIDE EFFECTS:
                              0000  ;
                              0000  ;       NONE
                              0000  ;
                              0000  ;--
                              0000          .ENTRY  -
                        0000  0000  LIB$ASSIGN,     ^M<>
           41'AF   6C    FA   0002          CALLG   (AP),B^LIB$ASSIGN_DEV           ;ATTEMPT DEVICE ASSIGNMENT
              16   50    E8   0006          BLBS    R0,10$                          ;BRANCH IF SUCCESSFUL
                              0009  ;
                              0009  ; PROCESS LIST OF RECOGNIZABLE FAILURES AS A DEVICE NAME THAT SHOULD BE
                              0009  ; RETURNED AS AN ERROR RATHER THAN ATTEMPTING AS A FILE NAME
                              0009  ;
     50    00000000'8F   D1   0009          CMPL    #RMS$_CHN,R0                    ;DEVICE ALLOCATED
                   0D    13   0010          BEQL    10$
     50    00000000'8F   D1   0012          CMPL    #RMS$_PRV,R0                    ;NO PRIVILEGE FOR DEVICE ACCESS
                   04    13   0019          BEQL    10$
```

```
                 20'AF    6C    FA 001B          CALLG    (AP),B^LIB$ASSIGN_FILE          ;ATTEMPT AS GENERAL FILE SPEC
                                04 001F  10$:    RET                                      ;RETURN STATUS FROM SUBROUTINE
                                   0020
                                   0020
                                   0020  ;
                                   0020  ; GET CHANNEL ASSOCIATED WITH FILE SPEC, ASSUME NO SPECIAL PUNCTUATION
                                   0020  ;
                                   0020          .ENTRY   -
                        007C       0020  LIB$ASSIGN_FILE,^M<R2,R3,R4,R5,R6>
          5E     000000B0 8F    C2 0022          SUBL     #FAB$C_BLN+NAM$C_BLN,SP         ;ALLOCATE ROOM FOR FAB AND NAM
                    56    5E    D0 0029          MOVL     SP,R6                           ;GET ADDRESS OF FAB
                        007C    30 002C          BSBW     ASSIGN_SETUP                    ;PERFORM COMMON SETUP
                                   002F          $PARSE   FAB=<(R6)>                      ;GET THE DEVICE INFORMATION
                       05 50    E9 0038  10$:    BLBC     R0,20$                          ;BRANCH IF ERROR TO RETURN CODE
               00F1'CF    6C    FA 003B          CALLG    (AP),W^ASSIGN_CHAN              ;CALL FOR STACK CLEANUP
                                04 0040  20$:    RET                                      ;CHANNEL
                                   0041
                                   0041  ;
                                   0041  ; GET CHANNEL ASSOCIATED WITH FILE SPEC, FORCE TO LOOK LIKE DEVICE NAME AND
                                   0041  ; ONLY SUCCEED IF INPUT REPRESENTS DEVICE NAME.
                                   0041  ;
                                   0041          .ENTRY   -
                        007C       0041  LIB$ASSIGN_DEV, ^M<R2,R3,R4,R5,R6>
          5E     000000B0 8F    C2 0043          SUBL     #FAB$C_BLN+NAM$C_BLN,SP         ;ALLOCATE ROOM FOR FAB AND NAM
                    56    5E    D0 004A          MOVL     SP,R6                           ;GET ADDRESS OF FAB
                        005B    30 004D          BSBW     ASSIGN_SETUP                    ;PERFORM COMMON SETUP
              61    50    3A    3A 0050  10$:    LOCC     #^A/:/,R0,(R1)                  ;DOES STRING CONTAIN A COLON?
                          11    13 0054          BEQL     20$                             ;BRANCH IF NO
                          50    D7 0056          DECL     R0                              ;WAS THIS LAST CHARACTER?
                          2A    13 0058          BEQL     30$                             ;BRANCH IF YES
                 01 A1    3A    91 005A          CMPB     #^A/:/,1(R1)                    ;IS THIS A NODE SPECIFIER?
                          24    12 005E          BNEQ     30$                             ;BRANCH IF NOT, EXPLICIT ":"
                    51    02    C0 0060          ADDL     #2,R1                           ;BUMP PAST "::"
                          50    D7 0063          DECL     R0                              ;ONE LESS CHARACTER IN COUNT
                          E9    11 0065          BRB      10$                             ;TRY FOR REAL ":"
                    50    62    9A 0067  20$:    MOVZBL   (R2),R0                         ;GET SIZE OF STRING
                    50    04    C0 006A          ADDL     #4,R0                           ;ROUND UP FOR STACK ALIGNMENT
                    50    03    CA 006D          BICL     #3,R0                           ;SIZE NEEDED ON STACK
                    5E    50    C2 0070          SUBL     R0,SP                           ;ALLOCATE SPACE FOR COPY
   6E    50  3A  04 B2    62    2C 0073          MOVC5    (R2),@4(R2),#^A/:/,R0,(SP)      ;MAKE A COPY WITH A ":" AT END
           34 A6 04 BC    01    81 007A          ADDB3    #1,@4(AP),FAB$B_FNS(R6)         ;SET SIZE WITH COLON
                 2C A6    5E    D0 0080          MOVL     SP,FAB$L_FNA(R6)                ;SET ADDRESS OF COPY
              00 04 A6    10    E2 0084  30$:    BBSS     #FAB$V_NFS,FAB$L_FOP(R6),35$    ;SET NON FILE STRUCTURE ACCESS
                                   0089  35$:    $PARSE   FAB=<(R6)>                      ;GET THE DEVICE INFORMATION
                       15 50    E9 0092  40$:    BLBC     R0,50$                          ;BRANCH IF ERROR TO RETURN CODE
          50         0000'8F    3C 0095          MOVZWL   #SS$_IVDEVNAM,R0                ;ASSUME FAILURE
                                D3 009A          BITL     #<NAM$M_EXP_DIR!NAM$M_EXP_NAME!-;CHECK FOR ANYTHING BUT DEVICE
                                   009B                   NAM$M_EXP_TYPE!NAM$M_EXP_VER!-
                                   009B                   NAM$M_GRP_MBR!NAM$M_WILDCARD>,-
        0084 C6  00080147 8F       009B                   NAM$L_FNB+FAB$C_BLN(R6)
                          05    12 00A3          BNEQ     50$                             ;BRANCH IF ANY OTHERS SET
               00F1'CF    6C    FA 00A5          CALLG    (AP),W^ASSIGN_CHAN              ;CALL FOR STACK CLEANUP
                                04 00AA  50$:    RET
                                   00AB
                                   00AB  ;
                                   00AB  ;       Routine used for common setup between ASSIGN_FILE and ASSIGN_DEV
                                   00AB  ;
```

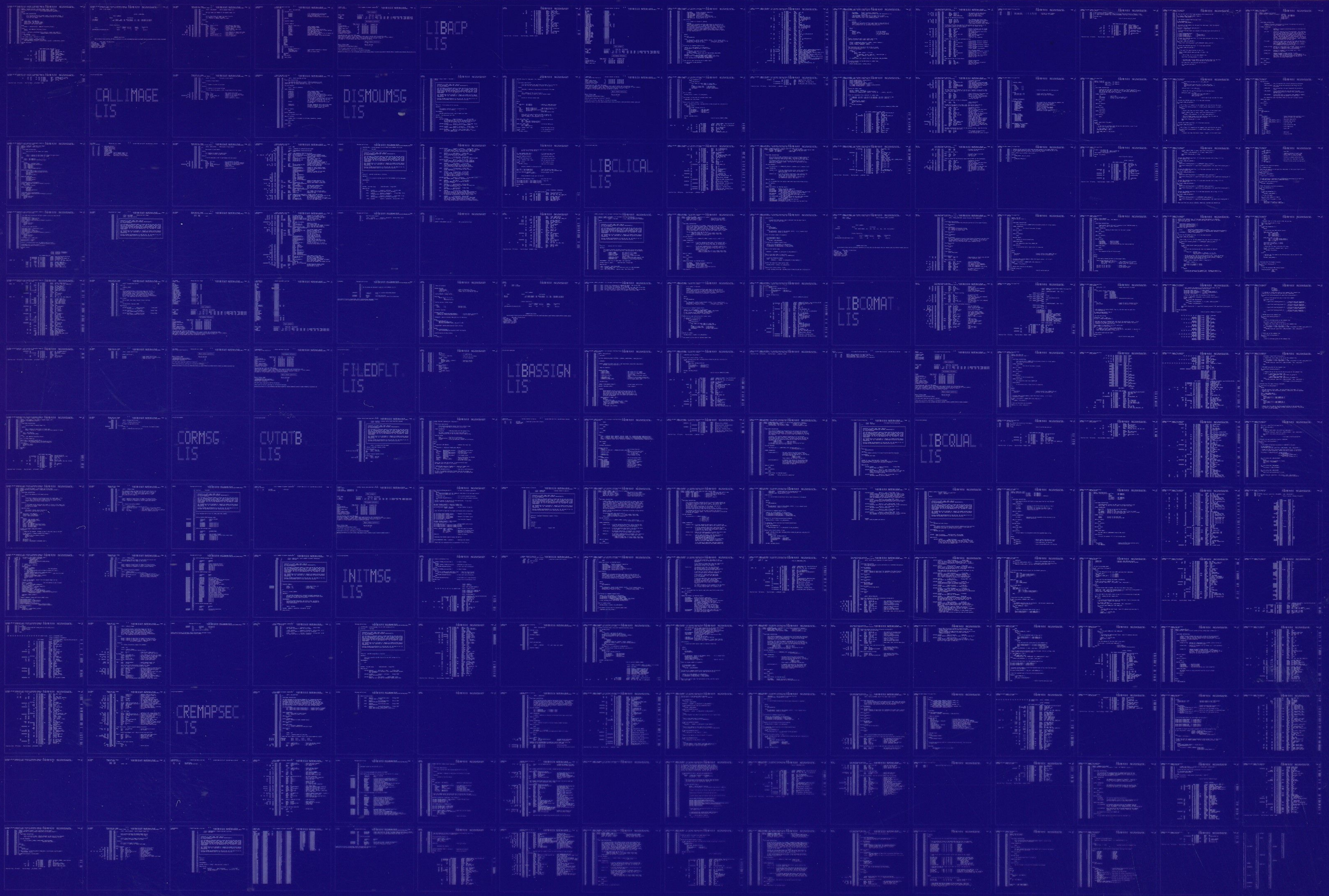
```
                                         00AB    190 ASSIGN_SETUP:
         50   00000000'8F     D0  00AB    191            MOVL    #LIB$_INVARG,R0                            ;ASSUME FAILURE
                     6C   02  91  00B2    192            CMPB    #2,(AP)                                    ;ARE THERE 2 ARGUMENTS?
                          39  12  00B5    193            BNEQ    10$                                        ;BRANCH IF NOT
            04 BC     00FF 8F  B1  00B7    194            CMPW    #255,@4(AP)                                ;IS SIZE OF DEVNAME REASONABLE?
                          31  19  00BD    195            BLSS    10$                                        ;BRANCH IF SIZE TOO BIG
   66    00B0 8F    00    6E  00  2C  00BF    196            MOVC5   #0,(SP),#0,#FAB$C_BLN+NAM$C_BLN,(R6) ;ZERO THEM
                     66   03  90  00C7    197            MOVB    #FAB$C_BID,FAB$B_BID(R6)                   ;SET BLOCK ID TO FAB
               01 A6    50 8F  90  00CA    198            MOVB    #FAB$C_BLN,FAB$B_BLN(R6)                   ;SET LENGTH OF FAB
                  50 A6    02  90  00CF    199            MOVB    #NAM$C_BID,NAM$B_BID+FAB$C_BLN(R6) ;SET BLOCK ID TO NAME BLOCK
               51 A6    60 8F  90  00D3    200            MOVB    #NAM$C_BLN,NAM$B_BLN+FAB$C_BLN(R6) ;SET LENGTH OF NAME BLOCK
               28 A6    50 A6  9E  00D8    201            MOVAB   FAB$C_BLN(R6),FAB$L_NAM(R6)                ;INDICATE NAME BLOCK EXISTS
                  52    04 AC  D0  00DD    202            MOVL    4(AP),R2                                   ;GET ADDRESS OF DESCRIPTOR
                     50   62  9A  00E1    203            MOVZBL  (R2),R0                                    ;GET SIZE OF STRING
                  51    04 A2  D0  00E4    204            MOVL    4(R2),R1                                   ;GET ADDRESS OF STRING
                  34 A6    50  90  00E8    205            MOVB    R0,FAB$B_FNS(R6)                           ;SET SIZE OF DEVICE NAME
                  2C A6    51  D0  00EC    206            MOVL    R1,FAB$L_FNA(R6)                           ;SET ADDRESS OF DEVICE NAME
                               05  00F0    207 10$:       RSB                                                ;RETURN
                                   00F1    208
                                   00F1    209 ;
                                   00F1    210 ;          Routine used for common closure for both ASSIGN_FILE and ASSIGN_DEV
                                   00F1    211 ;
                                   00F1    212 ASSIGN_CHAN:
                             0000  00F1    213            .WORD   0                                          ;USE COMMON REGISTER SET
                  54    65 A6  9E  00F3    214            MOVAB   NAM$T_DVI+1+FAB$C_BLN(R6),R4               ;SET UP ADDRESS OF DEVICE NAME FIELD
                           54  DD  00F7    215            PUSHL   R4                                         ;FORM A DESCRIPTOR
                  7E    FF A4  9A  00F9    216            MOVZBL  -1(R4),-(SP)                               ;GET SIZE OF FIELD
                     55    5E  D0  00FD    217            MOVL    SP,R5                                      ;R5 IS INPUT DESC FOR $TRNLOG
                  64    5F 8F  91  0100    218            CMPB    #^A\_\,(R4)                                ;DO WE ALREADY HAVE A PHYSICAL NAME?
                           30  13  0104    219            BEQL    10$                                        ;BRANCH IF YES TO ASSIGN THE CHANNEL
                     5E    32  C2  0106    220            SUBL    #50,SP                                     ;ALLOCATE A BUFFER
                           5E  DD  0109    221            PUSHL   SP                                         ;SET UP THE ADDRESS
                           32  DD  010B    222            PUSHL   #50                                        ;AND SIZE
                     53    5E  D0  010D    223            MOVL    SP,R3                                      ;R3 IS OUTPUT DESC FOR $TRNLOG
                                   0110    224            $TRNLOG_S (R5),(R3),(R3)                           ;TRANSLATE THE STRING
                     29 50     E9  0123    225            BLBC    R0,20$                                     ;BRANCH IF FAILURE
                     55    53  D0  0126    226            MOVL    R3,R5                                      ;SET UP TO USE TRANSLATED STRING
                  04 B5    1B  91  0129    227            CMPB    #27,@4(R5)                                 ;DOES OUTPUT START WITH ESCAPE?
                           07  12  012D    228            BNEQ    10$                                        ;BARNCH IF NOT
                     65    04  C2  012F    229            SUBL    #4,(R5)                                    ;REDUCE SIZE BY PROCESS PERM STUFF
                  04 A5    04  C0  0132    230            ADDL    #4,4(R5)                                   ;POINT PAST IT
                           00  DD  0136    231 10$:       PUSHL   #0                                         ;AREA TO GET CHANNEL BACK IN
                     52    5E  D0  0138    232            MOVL    SP,R2                                      ;GET ADDRESS OF AREA
                                   013B    233            $ASSIGN_S (R5),(R2)                                ;ASSIGN A CHANNEL
                     04 50     E9  0148    234            BLBC    R0,20$                                     ;BRANCH IF THIS METHOD FAILS ALSO
                  08 BC    62  B0  014B    235            MOVW    (R2),@8(AP)                                ;STORE THE CHANNEL FOR THE USER
                               04  014F    236 20$:       RET                                                ;RETURN
                                   0150    237
                                   0150    238            .END
```

```
                                                                   B  8
LIB$ASSIGN                        ; Assign channel to device                    16-SEP-1984 02:17:54  VAX/VMS Macro V04-00        Page   7
Symbol table                                                                     5-SEP-1984 04:40:09  [VMSLIB.SRC]LIBASSIGN.MAR;1      (4)

$$.TMP1                          = 00000001
$$.TMP2                          = 00000066
$$T1                             = 00000001
ASSIGN_CHAN                        000000F1 R       02
ASSIGN_SETUP                       000000AB R       02
FAB$B_BID                        = 00000000
FAB$B_BLN                        = 00000001
FAB$B_FNS                        = 00000034
FAB$C_BID                        = 00000003
FAB$C_BLN                        = 00000050
FAB$L_FNA                        = 0000002C
FAB$L_FOP                        = 00000004
FAB$L_NAM                        = 00000028
FAB$V_NFS                        = 00000010
LIB$ASSIGN                         00000000 RG      02
LIB$ASSIGN_DEV                     00000041 RG      02
LIB$ASSIGN_FILE                    00000020 RG      02
LIB$_INVARG                        ********    X    02
NAM$B_BID                        = 00000000
NAM$B_BLN                        = 00000001
NAM$C_BID                        = 00000002
NAM$C_BLN                        = 00000060
NAM$L_FNB                        = 00000034
NAM$M_EXP_DIR                    = 00000040
NAM$M_EXP_NAME                   = 00000004
NAM$M_EXP_TYPE                   = 00000002
NAM$M_EXP_VER                    = 00000001
NAM$M_GRP_MBR                    = 00080000
NAM$M_WILDCARD                   = 00000100
NAM$T_DVI                        = 00000014
RMS$_CHN                           ********    X    02
RMS$_PRV                           ********    X    02
SS$_IVDEVNAM                       ********    X    02
SYS$ASSIGN                         ********   GX    02
SYS$PARSE                          ********   GX    02
SYS$TRNLOG                         ********   GX    02

                                                  +----------------+
                                                  ! Psect synopsis !
                                                  +----------------+

PSECT name                      Allocation          PSECT No.  Attributes
----------                      ----------          ---------  ----------
.  ABS  .                       00000000  (    0.)  00 (  0.)  NOPIC   USR   CON   ABS   LCL NOSHR NOEXE NORD  NOWRT NOVEC BYTE
$ABS$                           00000000  (    0.)  01 (  1.)  NOPIC   USR   CON   ABS   LCL NOSHR   EXE   RD    WRT NOVEC BYTE
_LIB$CODE                       00000150  (  336.)  02 (  2.)    PIC   USR   CON   REL   LCL   SHR   EXE   RD  NOWRT NOVEC LONG

                                               +------------------------+
                                               ! Performance indicators !
                                               +------------------------+

Phase                      Page faults    CPU Time       Elapsed Time
-----                      -----------    --------       ------------
Initialization                      29    00:00:00.09    00:00:00.78
Command processing                 107    00:00:00.47    00:00:01.61
Pass 1                             171    00:00:03.96    00:00:08.75
Symbol table sort                    0    00:00:00.36    00:00:00.61
```

```
Pass 2                          57      00:00:00.90     00:00:01.86
Symbol table output              6      00:00:00.04     00:00:00.04
Psect synopsis output            2      00:00:00.02     00:00:00.02
Cross-reference output           0      00:00:00.00     00:00:00.00
Assembler run totals           374      00:00:05.85     00:00:13.69

The working set limit was 1050 pages.
21183 bytes (42 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 351 non-local and 12 local symbols.
238 source lines were read in Pass 1, producing 19 object records in Pass 2.
16 pages of virtual memory were used to define 14 macros.

                                        +----------------------------+
                                        ! Macro library statistics !
                                        +----------------------------+

Macro library name                      Macros defined
------------------                      --------------
_$255$DUA28:[SYSLIB]STARLET.MLB;2                11

486 GETS were required to define 11 macros.

There were no errors, warnings or information messages.

MACRO/DISA=TRACE/LIS=LIS$:LIBASSIGN/OBJ=OBJ$:LIBASSIGN MSRC$:LIBASSIGN/UPDATE=(ENH$:LIBASSIGN)
```

0435 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



LIBACP
LIS

CALLIMAGE
LIS

DISMOUMSG
LIS

LIBCLICAL
LIS

LIBCQMAT
LIS

FILEDFLT
LIS

LIBASSIGN
LIS

CORMSG
LIS

CVTATB
LIS

LIBCQUAL
LIS

INITMSG
LIS

CREMAPSEC
LIS